# Multimedia Systems Lecture – 30,31,32

*By*

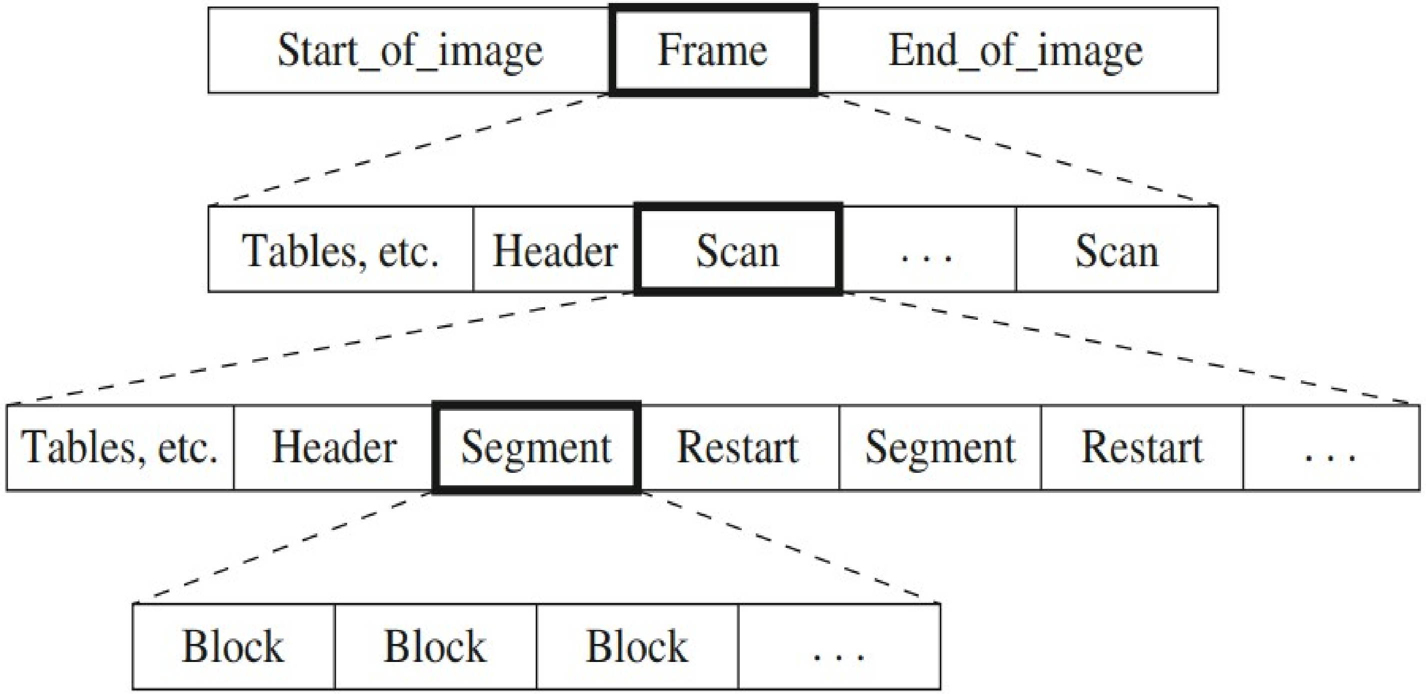
Dr. Priyambada Subudhi

Assistant Professor

IIIT Sri City

# JPEG Bitstream

- A ***frame*** is a picture
- A ***scan*** is a pass through the pixels (e.g., the red component).
- A ***segment*** is a group of blocks
- A ***block*** consists of 8 × 8 pixels



- ***Frame header***
  - Bits per pixel
  - (Width, height) of image
  - Number of components
  - Unique ID (for each component)
  - Horizontal/vertical sampling factors (for each component)
  - Quantization table to use (for each component).

- ***Scan header***
  - Number of components in scan
  - Component ID (for each component)
  - Huffman/Arithmetic coding table (for each component).

# Drawbacks of JPEG

- JPEG produced ***decent quality*** for the storage and transmission needs. Also, it has an ***acceptable computational complexity***, and the DCT computation can easily be ***implemented in hardware*** *for devices that involve image capture*.

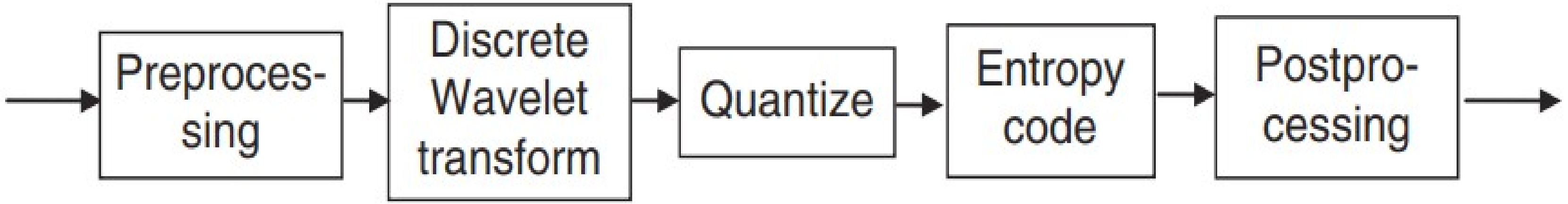
- However, today's needs are far greater than the provided functionality of JPEG, such as ***better compression requirements*** for ***larger imagery*** in a variety of applications, processing needs on compressed image bit streams, and so on.

- ***Poor low bit-rate compression***—JPEG offers excellent rate-distortion performance in the mid and high bit rates, but at ***low bit rates, the perceived distortion becomes unacceptable***.
- ***Lossy and lossless compression***—There is currently no standard that can provide superior ***lossless and lossy compression in a single coded stream.***
- ***Random access of the bit stream***—JPEG does ***not allow random access*** because each of the 8×8 blocks is interdependent. This prevents certain application scenarios where the end user might want to decode and see only a certain part of the image (or region of interest).
- ***Large image handling***—JPEG does not allow for the compression of images larger than ***64K by 64K*** without tiling.

- ***Single compression architecture***—The current JPEG standard has about 44 modes. Many of these are application specific and not used by the majority of decoders.
- ***Transmission in noisy environments***—JPEG was created before wireless communications became an everyday reality; therefore, it does not acceptably handle such an ***error-prone channel***.
- **Computer-generated images and documents**—JPEG was optimized for natural images and ***does not perform well on computer-generated images and document imagery***. This is because JPEG is well suited to continuous tone imagery but not constant tone or slow changing tone imagery.

# JPEG 2000

- The JPEG 2000 compression pipeline makes use of the ***Discrete Wavelet transform (DWT)*** to compress images.
- DWT is known to work better than the Discrete Cosine transform in the way it ***distributes energy among the frequency coefficients***.



## 1. THE PREPROCESSING STEP

It is responsible for ***tiling***, ***conversion into the YCrCb formats***, and ***level offsetting***.

- **Tiling:**
  - Tiling process partitions the image into **rectangular**, but **equal-sized** and **nonoverlapping blocks**.
  - Each tile is then independently processed for DWT analysis, quantization, entropy coding, and so on.
  - The tiling process is purely optional and is done only to reduce memory requirements, as needed to deal with very large images.
- **YCrCb conversion process:** It is similar to the JPEG pipeline.
- **Level offsetting process:** It refers to shifting the DC levels by subtracting a constant value from each pixel value in the image (or tiles).

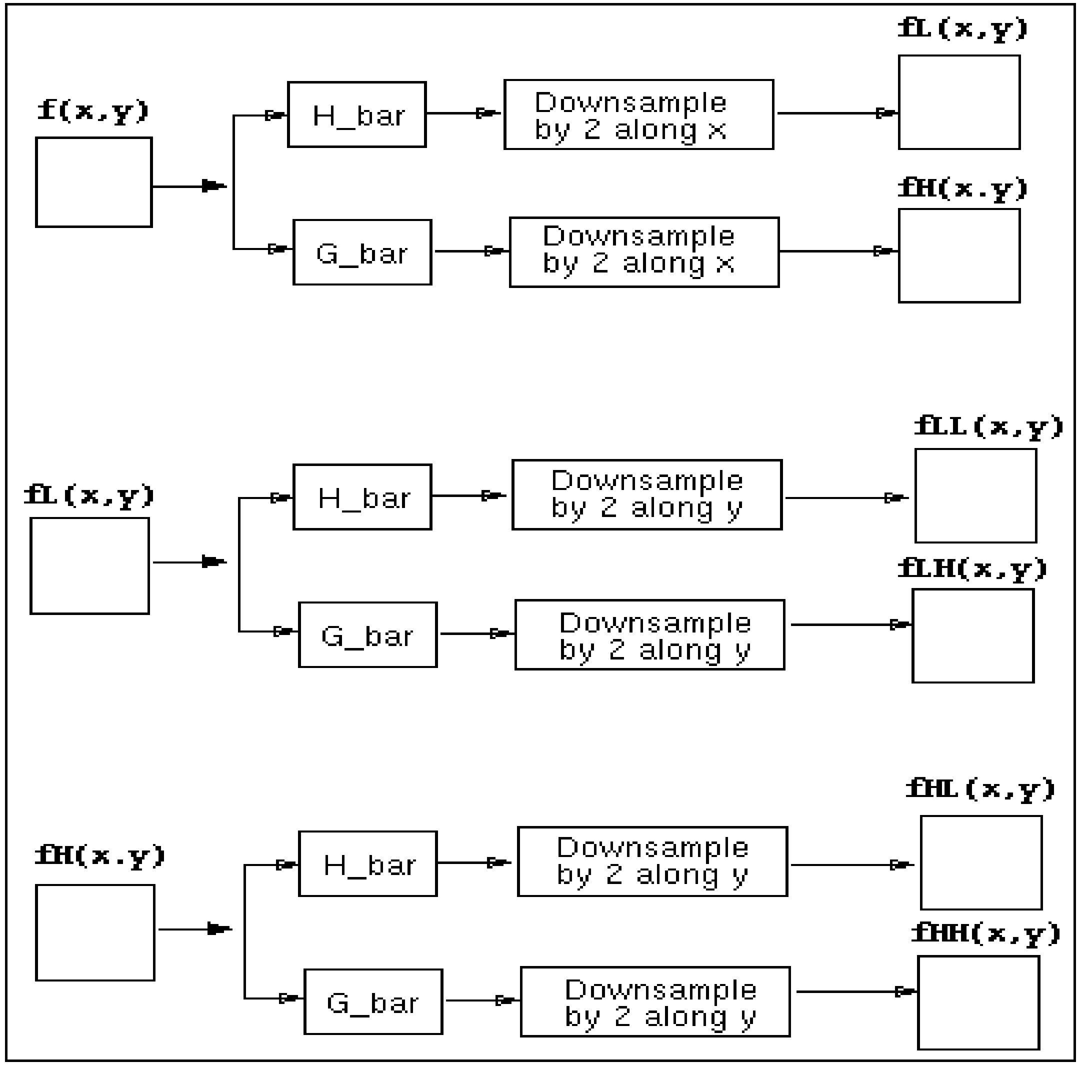
## 2. The Discrete Wavelet Transform

- It seeks to represent a signal with good resolution in ***both time and frequency***, by using a set of basis functions called **wavelets**.
- Wavelet transform coders process ***high and low frequency parts of image independently***
  - DCT methods have difficulties with high-frequency information
- Wavelet method ***transforms image as a whole*** (not subdivided into pixel blocks)
  - No blocking artifacts occur
  - Wavelet coders degrade gracefully

- Image is first filtered along the x dimension, resulting in low- pass and high-pass image

- Since bandwidth of both low pass and high pass image is now half that of the original image, both filtered images can be down-sampled by factor 2 without loss of information

- Then both filtered images are again filtered and down-sampled along the y dimension resulting in four sub-images



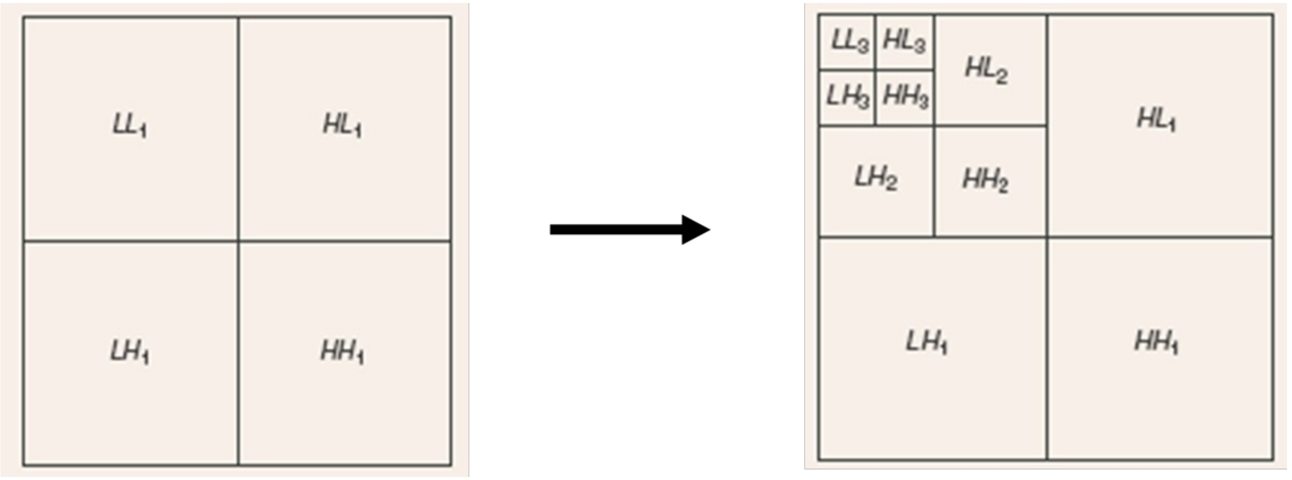
# Wavelet Transform

**LL**—Low subbands of the filtering in both dimensions, rows and columns

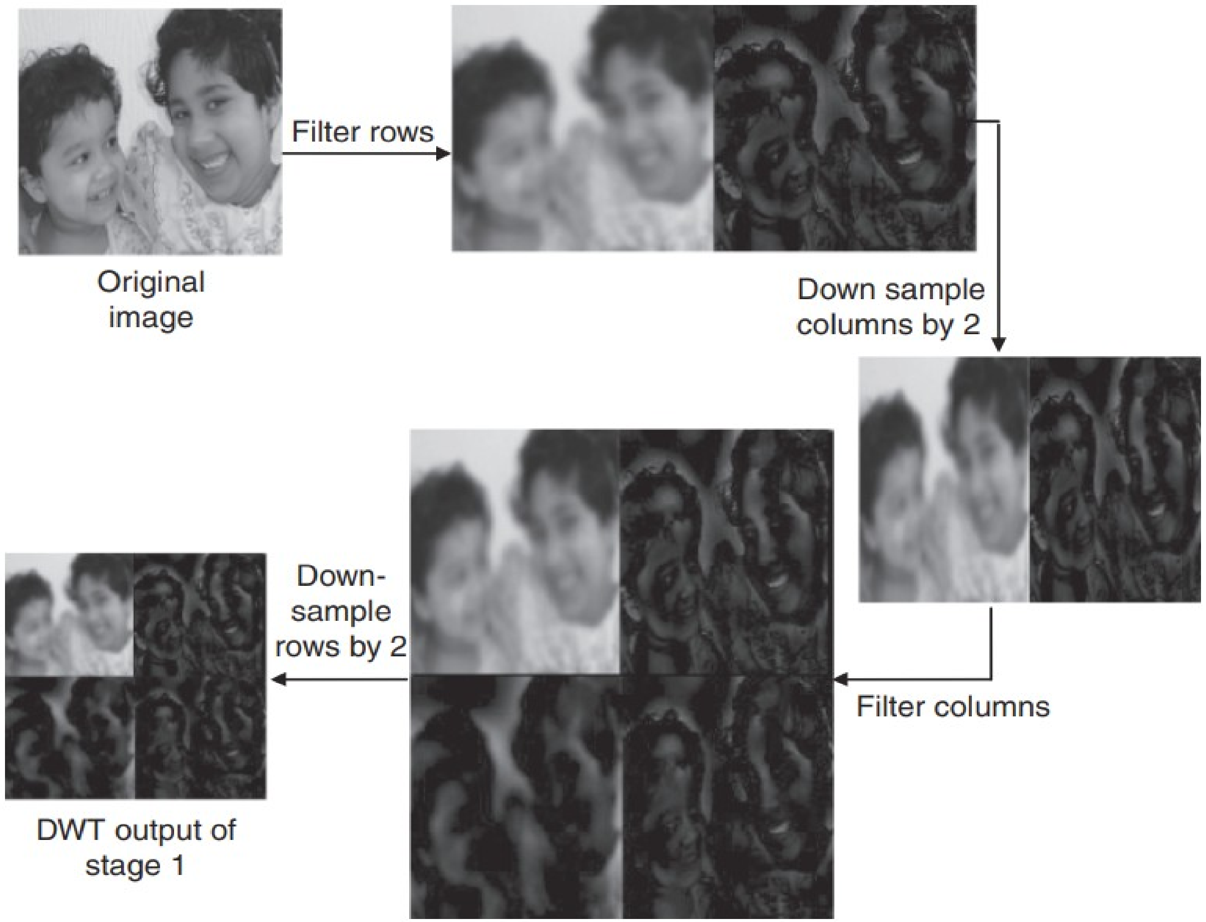
**HL**—High subbands after row filtering and low subbands after column filtering

**LH**—Low subbands after row filtering and high subbands after column filtering

**HH**—High subbands after both row and column filtering

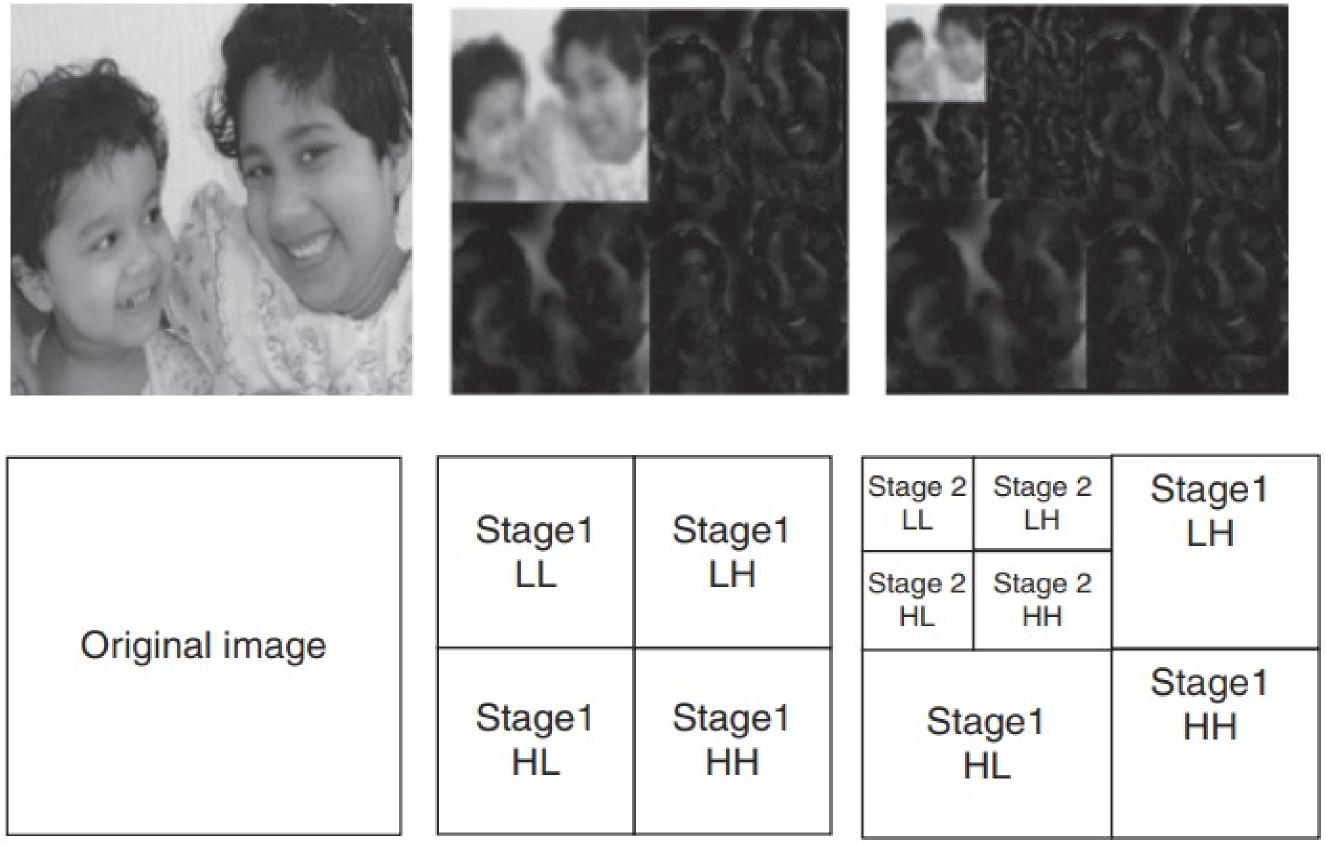


# DWT process for the Y component of the sample image used



- In JPEG 2000, every original input image signal goes through multiple stages of the DWT.
- The number of stages performed depends on the implementation.
- The second stage repeats the same process with the LL subband output of the previous stage.
- The higher subbands (HL, LH, and HH) hardly contain any significant samples and, therefore, only the LL subband is further transformed.
- Although JPEG 2000 supports from 0 to 32 stages, usually 4 to 8 stages are used for natural images.

Original input and the output of discrete wavelet processing at the first two levels. The top row shows the imaged outputs. The bottom row shows what each block at a level contains.



## 3. Qunatization

- After transformation, all coefficients are quantized using scalar quatization.
- Quantization reduces coefficients in precision. The operation is lossy unless the quatization step is 1 and the coefficients integers
- The process follows the formula:

Largest integer not exceeding $a_b$

$$q_b(u,v) = sign\,(a_b(u,v)) \left\lfloor \frac{|a_b(u,v)|}{\Delta_b} \right\rfloor$$

Quantized value

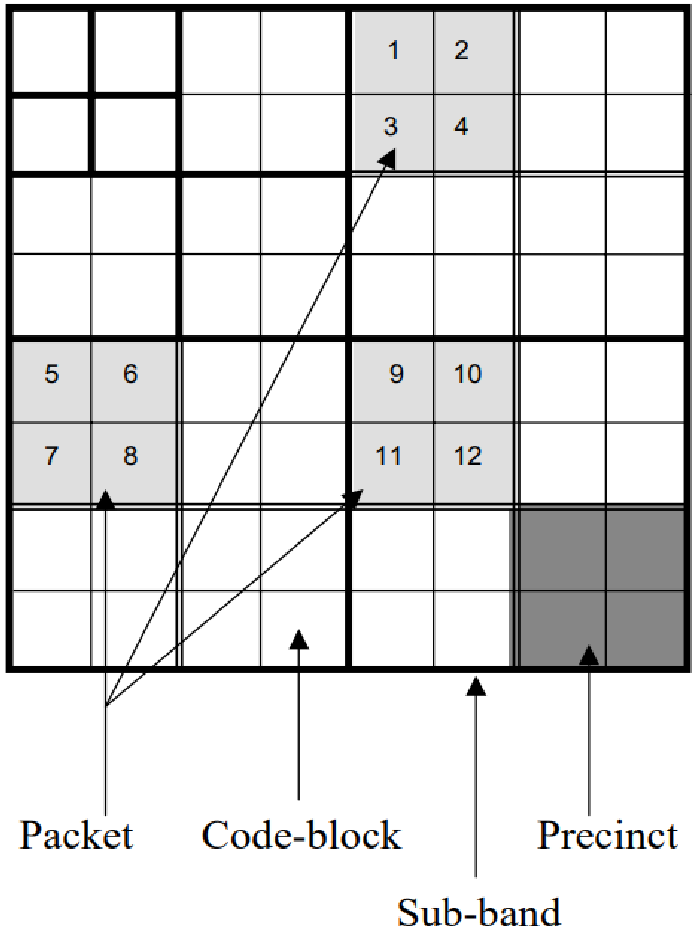
Transform coefficient of sub-band b

Quantization step

## 4. Encoding

- It is based on block-based encoding scheme known as Embedded Block Coding with Optimized Truncations (***EBCOT***).
- Each subband is partitioned in blocks that are encoded independently via the Embedded Block Coding with Optimized Truncations (EBCOT) algorithm.
- typically $64 \times 64$, or other size no less than $32 \times 32$.

- Code-blocks, precincts and packets



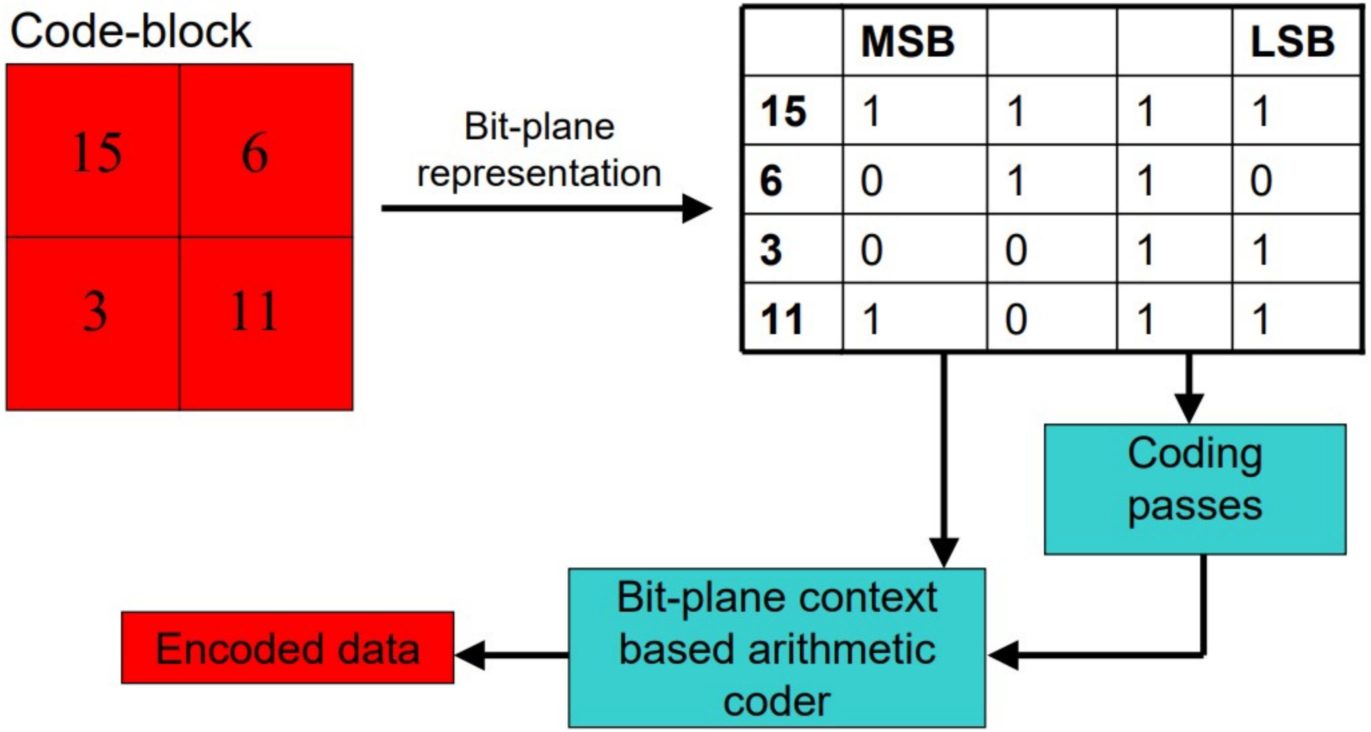

**Precinct:** each sub-band is divided into rectangular blocks called precincts.
**Packets:** three spatially consistent rectangles comprise a packet.
**Code-block:** each precinct is further divided into non-overlapping rectangles called code-blocks.

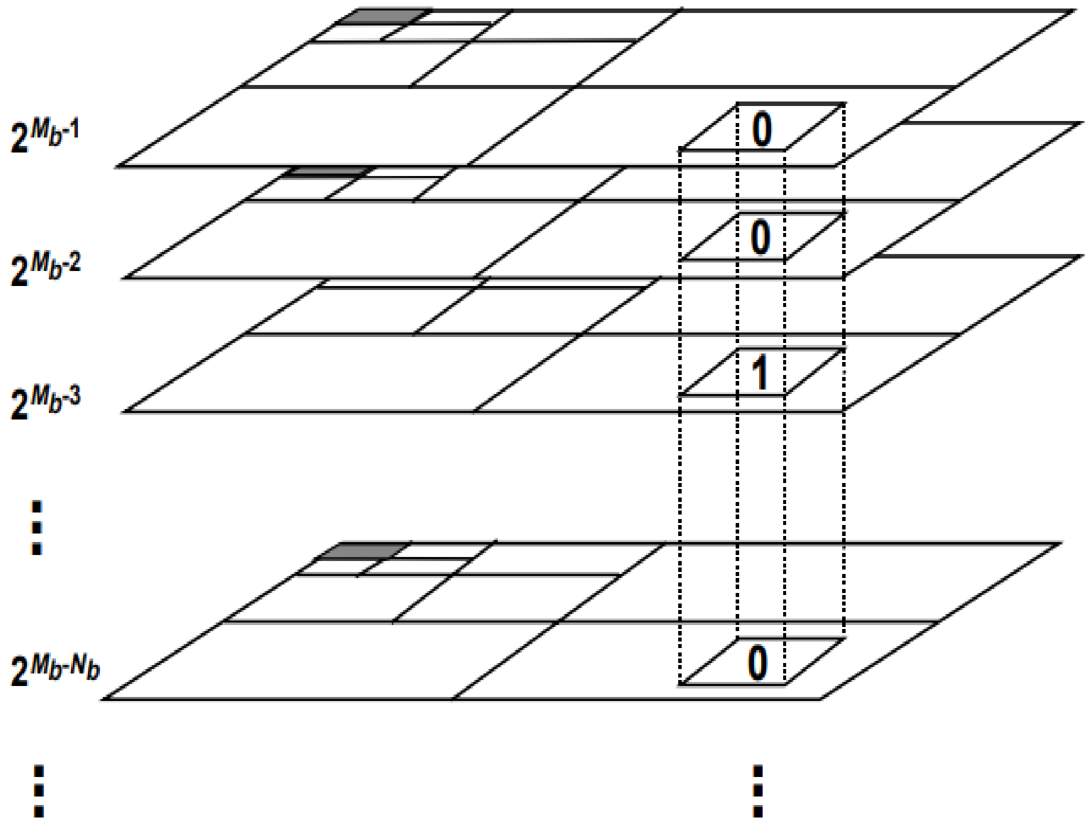
Each code-block forms the input to the entropy encoder and is encoded independently.

Within a packet, code-blocks are visited in raster order.

# Entropy Coding: Bit-planes

- The coefficients in a code block are separated into bit-planes. The individual bit-planes are coded in 1-3 coding passes.






- During this progressive bitplane encoding, a quantized wavelet coefficient is called ***insignificant*** if the quantizer index is still zero
- Once the first nonzero bit is encoded, the coefficient becomes ***significant***.
- Once a coefficient becomes significant, all subsequent bits are referred to as ***refinement*** bits.

Since the DWT packs most of the energy in the low frequency subbands, the majority of the wavelet coefficients will have low amplitudes. Consequently, many quantized indices will be insignificant in the earlier bitplanes, leading to a very low information content for those bitplanes.

# Entropy Coding: Coding Passes

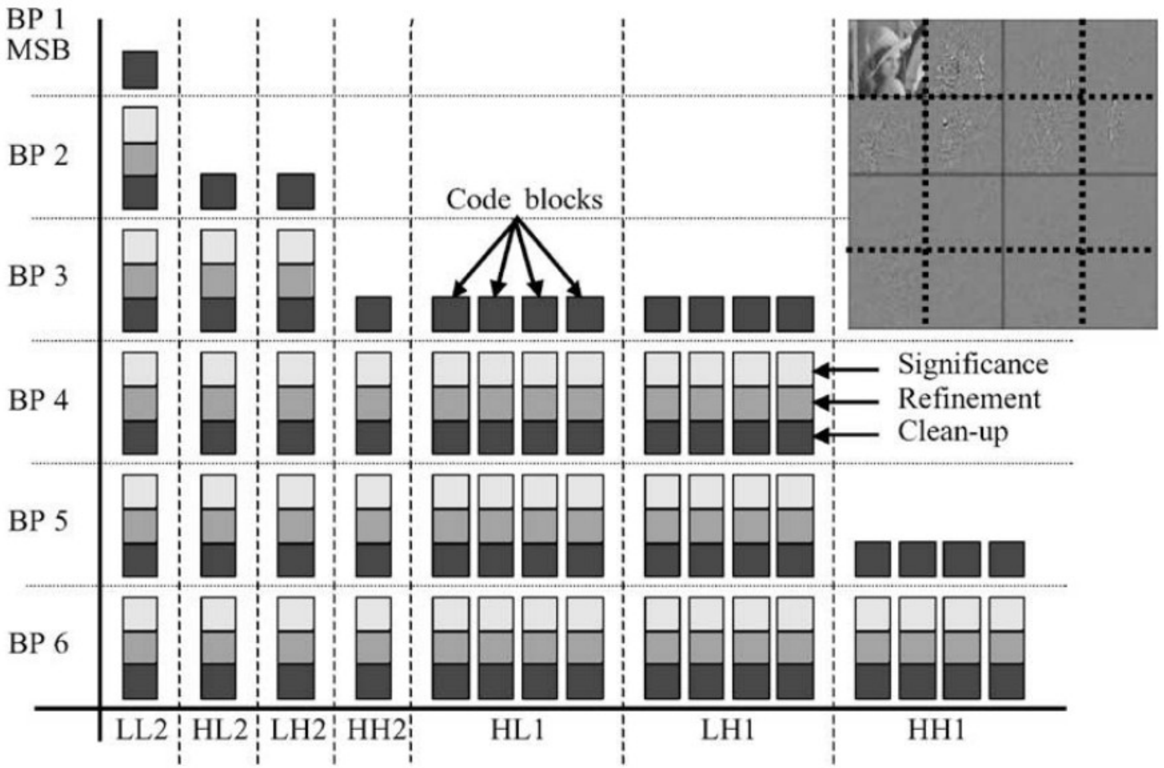
- Each of these coding passes collects contextual information about the bitplane data. The contextual information along with the bit-planes are used by the arithmetic encoder to generate the compressed bit-stream.
- The coding passes are:
  - ***Significance propagation pass:*** *T*he insignificant coefficients that have the highest probability of becoming significant, as determined by their immediate eight neighbors, are encoded.
  - ***Magnitude refinement pass:*** *T*he significant coefficients are refined by their bit representation in the current bitplane.
  - ***Clean-up pass:*** *A*ll the remaining coefficients in the bitplane are encoded as they have the lowest probability of becoming significant..

# Entropy Coding: MQ Coder

- JPEG2000 uses an efficient coding method for exploiting the redundancy of the bit-planes known as context-based adaptive binary arithmetic coding (***MQ Coder***).
- An adaptive binary arithmetic coder can be viewed as an encoding device that accepts the binary symbols in a source sequence, along with their corresponding probability estimates, and produces a codestream with a length at most two bits greater than the combined ideal codelengths of the input symbols.
- Adaptivity is provided by updating the probability estimate of a symbol based upon its present value and history.

- In JPEG2000, the probability of a binary symbol is estimated from a ***context*** formed from its current ***significance state*** as well as the significance states of its immediate eight neighbors (256 different contexts, only 9 considered) as determined from the previous bitplane and the current bitplane, based on coded information up to that point.
- if a coefficient is found to be significant, its sign needs to be encoded. The sign value is also arithmetic encoded using five contexts that are determined from the significance and the sign of the coefficient's four horizontal and vertical neighbors.
- A special mode, referred to as the ***run mode***, is used to aggregate the coefficients that have the highest probability of remaining insignificant. More specifically, a run mode is entered if all the four samples in a vertical column of the stripe have insignificant neighbors.
- An encoded value of zero implies insignificance for all four samples, while an encoded value of one implies that at least one of the four samples becomes significant in the current bitplane.
- Each codeblock employs its own MQ-coder to generate a single arithmetic codeword for the entire codeblock.

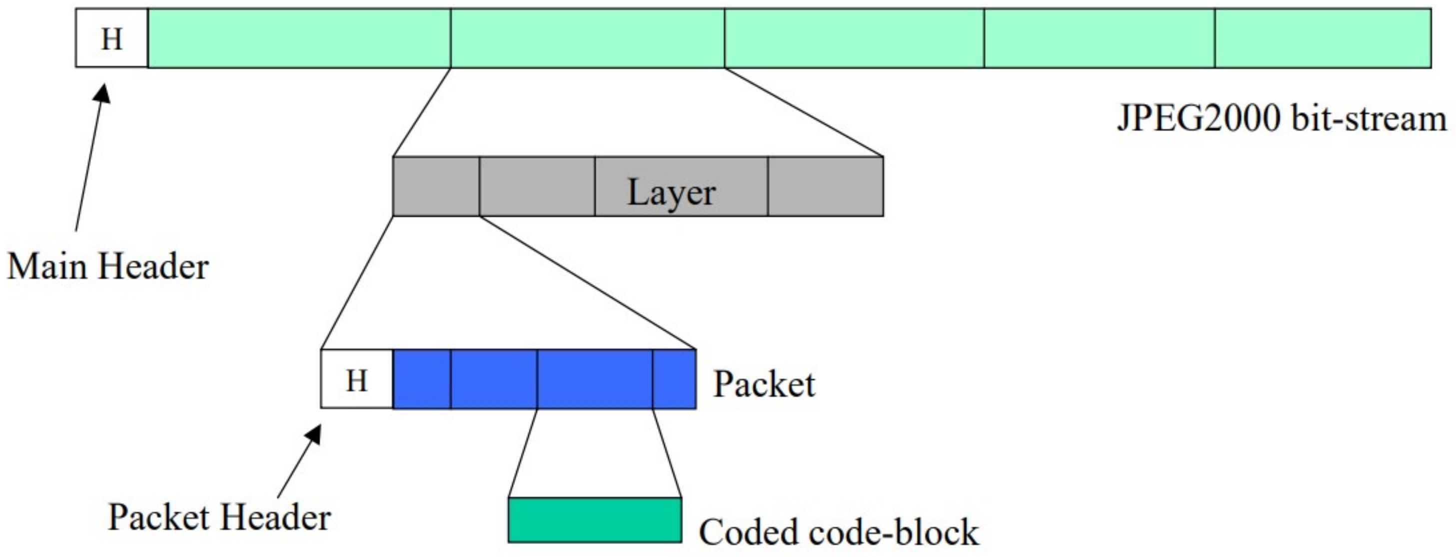
- The example image  of size 256×256 with two levels of decomposition, and the codeblock size is 64×64. Each square box in the figure represents the compressed data associated with a single coding pass of a 22 single codeblock.



- The arithmetic coding of the bitplane data is referred to as tier-1 (T1) coding.
- JPEG2000 organizes the compressed data from the codeblocks into units known as packets and layers during the tier-2 coding step.
- For each precinct, the compressed data for the codeblocks is first organized into one or more packets.
- A packet is the fundamental building block in a JPEG2000 codestream. Each packet starts with a packet header.
  - The packet header contains information about the number of coding passes for each codeblock in the packet.
  - Also contains the length of the compressed data for each codeblock.
- Packets from each precinct at all resolution levels in a tile are then combined to form layers.

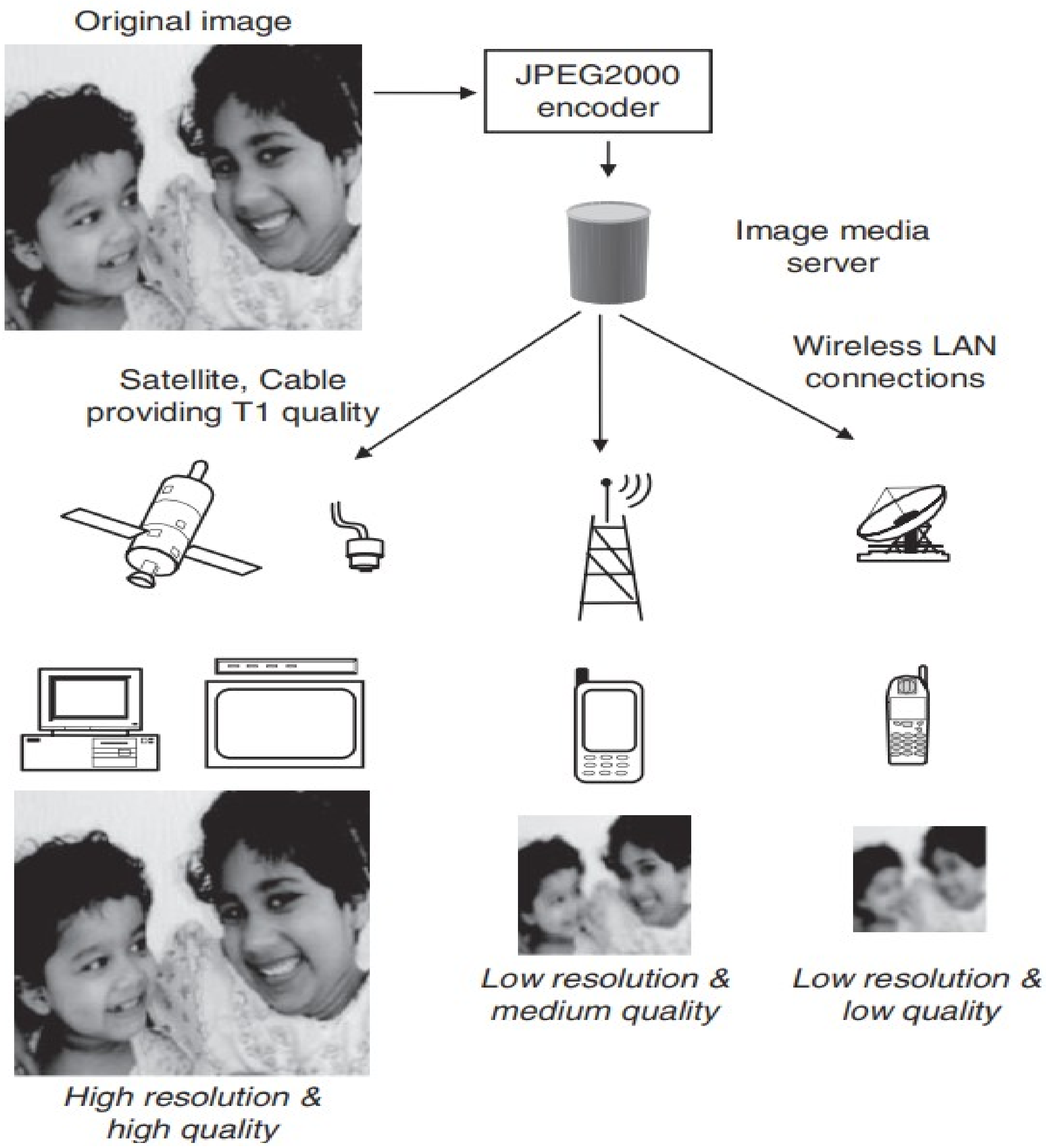
# JPEG2000 Bit-stream

- For each code-block, a separate bit-stream is generated.
- The coded data of each code-block is included in a packet.
- If more than one layer is used to encode the image information, the code-block bit-streams are distributed across different packets corresponding to different layers.



# JPEG 2000 Versus JPEG

- Encode once—platform-dependent decoding:
  - In JPEG 2000, the encoder decides the maximum resolution and image quality to be produced—from the highly compressed to completely lossless.
  - Because of the hierarchical organization of each band's coefficients, any image quality can be decompressed from the resulting bit stream.
  - Additionally, the organization of the bit stream also makes it possible to perform random access by decompressing a desired part of the image or even a specific image component.
  - Unlike JPEG, it is not necessary to decode and decompress the entire bit stream to display section(s) of an image or the image itself at various resolutions.


Original image
JPEG2000
encoder
Image media
server
Satellite, Cable
providing T1 quality
Wireless LAN
connections
High resolution &
high quality
Low resolution &
medium quality
Low resolution &
low quality

- Region-of-interest encoding—
  - Region-of-interest (ROI) coding pertains to coding a specific region with higher quality compared with the rest of the image.
  - This process can be predefined, or can change dynamically.
  - Predefined ROIs are normally set at encoding time, whereas dynamic ROIs are used in applications where specific regions of the image are chosen for decoding at finer resolutions compared with other parts of the image.
  - JPEG 2000 elegantly allows for ROI control by specifying a parameter known as an ***ROI mask***. The ROI mask informs the encoder (if static selection) or the decoder (if dynamic selection) about the range of wavelet coefficients that contribute to a region's reconstruction. These coefficients can be decoded first, before all the background is decoded.

- Working with compressed images—
    - Normally, imaging operations such as simple geometrical transformations (cropping, flipping, rotating, and so on) and filtering (using the frequency domain) are performed on the pixel domain representation of the image.
    - If the image is compressed, it is decompressed, and recompressed after the required operation.
    - However, with JPEG 2000, such operations can be applied to the compressed representation of the image.
- High compression (especially at low bitrates) with better quality
- Error resilience (robustness to bit errors when communication or storage devices are unreliable)
- Support for tiling